## CLAIMS:

What is claimed is:

1. A method of processing performance information in a data processing system, comprising the steps of:

receiving an interrupt signal at an interrupt unit of a processor of the data processing system;

determining if at least one of a pre handler routine and a post handler routine are enabled for an interrupt;

invoking the pre handler routine to record events at a first instant if the pre routine is enabled;

invoking an interrupt handler routine; and invoking the post handler routine to record events at a second instant if the post handler routine is enabled.

- 2. The method of claim 1, wherein recording events includes recording a plurality of counts.
- 3. The method of claim 1, wherein recording events includes recording a timestamp.
- 4. The method of claim 2, wherein the count represents the number of times an event occurs.
- 5. The method of claim 4, wherein the event is selected from the group consisting of cache misses and number of instructions executed.

- 6. The method of claim 1, wherein the first and second instants are associated with first and second timestamps, respectively.
- 7. The method of claim 1, further comprising a plurality of pre handler routines and a plurality of post handler routines, wherein each pre handler routine and each post handler routine records a different event on the occurrence of an interrupt.
- 8. The method of claim 1, wherein recording events includes accumulating a total value of counts.
- 9. The method of claim 8, wherein the total value of counts is accumulated by adding counts of events recorded.
- 10. The method of claim 8, wherein the total value of counts is displayed in a performance analysis tool.
- 11. The method of claim 2, wherein the count is not updated when the pre or post handler routine is invoked.
- 12. A method of executing instructions in a data processing system, comprising the steps of:

receiving an interrupt signal at an interrupt unit of a processor of the data processing system;

determining if at least one of a pre handler routine and a post handler routine are enabled for an interrupt;

invoking the pre handler routine to log a trace
record at a first instant if the pre routine is enabled;
invoking an interrupt handler routine;

invoking the post handler routine to log a trace record at a second instant if the post handler routine is enabled.

- 13. The method of claim 12, wherein the trace record includes a from address of an instruction indicating where the interrupt occurs.
- 14. The method of claim 12, wherein the trace record includes a plurality of counts.
- 15. The method of claim 14, wherein the count represents the number of times an event occurs.
- 16. The method of claim 15, wherein the event is selected from the group consisting of cache misses and clock cycles.
- 17. The method of claim 12, wherein the trace record includes a timestamp.
- 18. The method of claim 12, further comprising a plurality of pre handler routines and a plurality of post handler routines, wherein each pre handler routine and each post handler routine logs a different event on the occurrence of an interrupt.

- 19. The method of claim 12, wherein the first and second instants are associated with first and second timestamps, respectively.
- 20. The method of claim 12, wherein the pre handler routine or the post handler routine monitors a count of recorded events to determine if an overflow occurred.
- 21. The method of claim 20, wherein the pre handler routine or the post handler routine handles the overflow by reading and resetting the count.
- 22. A data processing system, comprising: an interrupt unit for receiving interrupt signals; a pre handler routine and a post handler routine; wherein responsive to receiving an interrupt signal for an interrupt at the interrupt unit, the pre handler routine logs a trace record at a first instant; and

wherein responsive to completion of the interrupt, the post handler routine logs a trace record at a second instant.

- 23. The system of claim 22, wherein the trace record includes a from address of an instruction indicating where the interrupt occurs.
- 24. The system of claim 22, wherein the trace record includes a plurality of counts.

- 25. The system of claim 24, wherein the count represents the number of times an event occurs.
- 26. The system of claim 25, wherein the event is selected from the group consisting of cache misses and clock cycles.
- 27. The system of claim 22, wherein the trace record includes a timestamp.
- 28. The system of claim 22, further comprising a plurality of pre handler routines and a plurality of post handler routines, wherein each pre handler routine and each post handler routine logs a different event on the occurrence of an interrupt.
- 29. The system of claim 22, wherein the first and second instants are associated with first and second time stamps, respectively.
- 30. The system of claim 24, wherein the count is not updated when the pre or post handler routine is invoked.
- 31. The system of claim 22, wherein the pre handler routine or the post handler routine monitors a count of recorded events to determine if an overflow occurred.
- 32. The system of claim 31, wherein the pre handler routine or the post handler routine handles the overflow by reading and resetting the count.

33. A computer program product in a computer readable medium, comprising:

first instructions for receiving an interrupt signal at an interrupt unit of a processor of the data processing system;

second instructions for determining if at least one of a pre handler routine and a post handler routine are enabled for an interrupt;

third instructions for invoking the pre handler routine to log a trace record at a first instant if the pre routine is enabled;

fourth instructions for invoking an interrupt handler routine;

fifth instructions for invoking the post handler routine to log a trace record at a second instant if the post handler routine is enabled.

- 34. The computer program product of claim 33, wherein the trace record includes a from address of an instruction indicating where the interrupt occurs.
- 35. The computer program product of claim 33, wherein the trace record includes a plurality of counts.
- 36. The computer program product of claim 35, wherein the count represents the number of times an event occurs.

- 37. The computer program product of claim 36, wherein the event is selected from the group consisting of cache misses and clock cycles.
- 38. The computer program product of claim 33, wherein the trace record includes a timestamp.
- 39. The computer program product of claim 33, further comprising a plurality of pre handler routines and a plurality of post handler routines, wherein each pre handler routine and each post handler routine logs a different event on the occurrence of an interrupt.
- 40. The computer program product of claim 33, wherein the first and second instants are associated with first and second timestamps in the trace record, respectively.
- 41. The computer program product of claim 33, wherein the pre handler routine or the post handler routine monitors a count of recorded events to determine if an overflow occurred.
- 42. The computer program product of claim 35, wherein the count is not updated when the pre or post handler is invoked.
- 43. The computer program product of claim 41, wherein the pre handler routine or the post handler routine handles the overflow by reading and resetting the count.

44. A method of executing branch instructions in a data processing system, comprising the steps of:

executing a branch instruction of a program;
receiving a signal at an interrupt unit of a
processor of the data processing system in response to
executing a trap, wherein the trap is executed in
response to executing the branch instruction of the
program.

invoking a pre handler routine to log a trace record at a first instant in response to receiving the signal;

invoking the post handler routine to log a trace record at a second instant when the execution of the branch instruction of a program is complete.

- 45. The method of claim 44, wherein the trace record includes a from address indicating an address of the branch instruction.
- 46. The method of claim 44, wherein the trace record includes a to address indicating an address of branch to instruction.
- 47. The method of claim 44, wherein the trace record includes a plurality of counts.
- 48. The method of claim 47, wherein the count represents the number of times an event occurs.
- 49. The method of claim 44, wherein the trace record includes a timestamp.

- 50. The method of claim 48, wherein the event is selected from the group consisting of cache misses and clock cycles.
- 51. The method of claim 44, further comprising a plurality of pre handler routines and a plurality of post handler routines, wherein each pre handler routine and each post handler routine logs a different event.
- 52. The method of claim 44, wherein the first and second instants are associated with first and second timestamps in the trace record, respectively.
- 53. The method of claim 47, wherein the count is not updated when the pre and post handler routine is invoked.